Listing of the Claims

The following listing of claims will replace all prior versions and listings of the claims in the application:

1. (Currently Amended) A method for processing a plurality of swap requests comprising:

receiving a first swap request in a pipeline in a first clock cycle, wherein the first swap request requests swapping active contents of a active register window with a first contents from a first register;

executing the first swap request including:

executing a first save operation in the first clock cycle, wherein the active contents of the active register window is saved to a corresponding register; and

executing a first restore operation in a second clock cycle, wherein the second clock cycle immediately follows the first clock cycle, wherein the first contents of the first register are restored to the active register window;

receiving a second swap request in the pipeline in the second clock cycle ~~immediately subsequent to the first swap request~~, wherein the second swap request requests swapping the first contents in the active register window with a second contents from a second register;

determining if the first register is a same register as the second register; and

executing the second swap request if the first swap request and the second swap request do not swap the same register, wherein executing the second swap request includes:

executing a second save operation in the second clock cycle, wherein the first contents of the active register window is saved to first register at substantially simultaneously with the executing the first restore operation; and

executing a second restore operation in a third clock cycle, wherein the third clock cycle immediately follows the second clock cycle, wherein the second contents of the second register are restored to the active register window, wherein the first swap request and the second swap request have a constant latency.

2. (Canceled)

3. (Canceled)

4. (Canceled)

5. (Previously Presented) The method of claim 1, further comprising:

delaying execution of the second swap request if the first register is the same register as the second register; and

executing the second swap request.

6. (Previously Presented) The method of claim 5, wherein the execution of the second swap request is delayed sufficiently to allow the execution of the first swap request to be completed.

7. (Original) The method of claim 5, wherein the execution of the second swap request is delayed a predetermined number of clock cycles.

8. (Original) The method of claim 5, wherein the execution of the second swap request is delayed one clock cycle.

9. (Original) The method of claim 1, wherein the pipeline includes more than one processing thread.

10. (Previously Presented) The method of claim 9, wherein determining if the first register is the same register as the second register includes determining if the first register in the corresponding processing thread is the same register as the second register in the corresponding processing thread.

11. (Previously Presented) The method of claim 1, wherein determining if the first register is the same register as the second register occurs as the second swap request is received.

12. (Currently Amended) A method for processing a plurality of consecutive swap requests in a multithreaded microprocessor pipeline comprising:

receiving a first swap request in a pipeline in a first clock cycle, wherein the first swap request requests swapping active contents of a active register window with a first contents from a first register;

executing the first swap request including:

executing a first save operation in the first clock cycle, wherein the active contents of the active register window is saved to corresponding register; and

executing a first restore operation in a second clock cycle, wherein the second clock cycle immediately follows the first clock cycle, wherein the first contents of the first register are restored to the active register window;

receiving a second swap request in the pipeline in the second clock cycle, wherein the second swap request requests swapping the first contents in the active register window with a second contents from a second register;

determining if the first register in the corresponding processing thread is a same register as the second register in the corresponding processing thread; and

executing the second swap request if the first swap request and the second swap request do not swap the same register, wherein executing the second swap request includes:

executing a second save operation in the second clock cycle, wherein the first contents of the active register window is saved to first register at substantially simultaneously with the executing the first restore operation; and

executing a second restore operation in a third clock cycle; wherein the third clock cycle immediately follows the second clock cycle, wherein the second contents of the second register are restored to the active register window, wherein the first swap request and the second swap request have a constant latency.

13. (Canceled)

14. (Previously Presented) The method of claim 12, further comprising:

delaying execution of the second swap request at least one clock cycle if the first register in the corresponding processing thread is the same register as the second register in the corresponding processing; and executing the second swap request.

15. (Currently Amended) A pipeline architecture for a processing thread comprising:

a plurality of pipeline registers, at least one of the plurality of pipeline registers being capable of comparing a first swap request and a second swap request;

a plurality of active registers;

logic for receiving a first swap request in the pipeline in a first clock cycle, wherein the first swap request requests swapping active contents of a first active register window of the plurality of active register windows in the pipeline with a first contents from a first register;

logic for executing the first swap request including:

executing a first save operation in the first clock cycle, wherein the active contents of the first active register window is saved to corresponding register; and

executing a first restore operation in a second clock cycle, wherein the second clock cycle immediately follows the first clock cycle, wherein the first contents of the first register are restored to the first active register window;

logic for receiving a second swap request in the pipeline in the second clock cycle ~~immediately subsequent to the first swap request~~, wherein the second swap request requests swapping the first contents in the first active register window with a second contents from a second register;

logic for determining if the first register is a same register as the second register; and logic for executing the second swap request if the first swap request and the second swap request do not swap the same register, wherein executing the second swap request includes:

executing a second save operation in the second clock cycle, wherein the first contents of the active register window is saved to first register at substantially simultaneously with the executing the first restore operation; and

executing a second restore operation in a third clock cycle, wherein the third clock cycle immediately follows the second clock cycle, wherein the second contents of the second register are restored to the active register window, wherein the first swap request and the second swap request have a constant latency.

16. (Original) The pipeline architecture of claim 15, wherein the plurality of pipeline registers includes at least eight pipeline registers, and wherein the at least eight pipeline registers are linked to one of the plurality of active registers.

17. (Original) The pipeline architecture of claim 15, wherein the plurality of pipeline registers includes 32 pipeline registers.

18. (Original) The pipeline architecture of claim 15, wherein the pipeline architecture is one of at least two pipeline architectures in a single multithreaded microprocessor.